

Applic. No.: 10/662,627  
Amdt. Dated December 14, 2005  
Reply to Office action of September 15, 2005

RECEIVED  
CENTRAL FAX CENTER

DEC 14 2005

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of claims:

Claim 1 (currently amended). A method of modular multiplication of a multiplicand by a multiplier within a cryptographic algorithm, in which a modulus is employed, wherein the multiplicand, the multiplier, and the modulus are parameters in the cryptographic algorithm, making use of using a multiplication look-ahead process and a reduction look-ahead process, ~~said~~ the method comprising the steps of:

transforming the modulus into a transformed modulus ~~that is~~ being greater than the modulus by multiplying the modulus by a transforming number, the transforming number being calculated using the modulus such that with a predetermined fraction of the transformed modulus having has a higher-order digit with a first predetermined value that is followed by at least one lower-order digit having a second predetermined value;

~~iterative~~ iteratively working off of the modular multiplication ~~making use of using~~ the multiplication look-ahead process and the reduction look-ahead process and utilizing the transformed modulus so as to obtain at the end

Applic. No.: 10/662,627  
Amdt. Dated December 14, 2005  
Reply to Office action of September 15, 2005

of the iteration a transformed result for the modular multiplication, the predetermined fraction of the transformed modulus being used in the reduction look-ahead process; and

re-transforming the transformed result by modular reduction of the transformed result utilizing the modulus.

Claim 2 (currently amended). [[A]] The method according to claim 1, wherein the step of ~~iterative~~ iteratively working off comprises a plurality of iteration steps, with a multiplication intermediate result and a reduction shift value being determined in one of ~~said~~ the iteration steps, with the reduction shift value being computed using a determination of the number of digits between the higher-order digit with the first predetermined value of the transformed modulus and the highest-order digit of the intermediate result having ~~said~~ the first predetermined value.

Claim 3 (currently amended). [[A]] The method according to claim 2, ~~wherein~~ which further comprises determining a multiplication shift value ~~is determined~~ in ~~said~~ the multiplication look-ahead process, and ~~wherein a~~ calculating the reduction shift value for the reduction look-ahead process ~~is calculated~~ by subtraction of ~~said~~ the predetermined number of digits from the multiplication shift value.

Applic. No.: 10/662,627.

Amdt. Dated December 14, 2005

Reply to Office action of September 15, 2005

Claim 4 (currently amended). [[A]] The method according to ~~any of the preceding claims~~ claim 1, wherein ~~said the~~ step of iterative iteratively working off comprises the following steps:

in a first iteration step:

(a) performing [[a]] the multiplication look-ahead process to obtain a multiplication shift value;

(b) multiplying a base raised to the power of the multiplication shift value by a current intermediate result to obtain a shifted intermediate result;

(c) performing [[a]] the reduction look-ahead process to obtain a reduction shift value by determining an auxiliary shift value equal to the number of digits between the higher-order digit with the first predetermined value of the predetermined fraction of the transformed modulus and the highest-order digit of the intermediate result having ~~said the~~ the first predetermined value, and by calculating the reduction shift value using the auxiliary shift value and the multiplication shift value;

Applic. No.: 10/662,627

Amdt. Dated December 14, 2005

Reply to Office action of September 15, 2005

(d) multiplying the transformed modulus by the base raised to the power of the reduction shift value to obtain a shifted transformed modulus; and

(e) summing the intermediate result and the multiplicand and subtracting the shifted transformed modulus to obtain an updated intermediate result.

Claim 5 (currently amended). [[A]] The method according to claim 1, wherein said predetermined fraction of the modulus is  $2/3$ .

Claim 6 (currently amended). [[A]] The method according to claim 5, wherein the multiplicand, the multiplier and the modulus are binary, with the base being 2, and wherein the higher-order digit of the predetermined fraction of the transformed modulus has [[a]] the first predetermined value of 1 and the at least one low-order digit has [[a]] the second predetermined value of 0.

Claim 7 (currently amended). [[A]] The method according to claim 6, wherein the most significant bit of the transformed modulus is a sign bit, and a higher-order section of the predetermined fraction of the modulus reads as follows:

Applic. No.: 10/662,627

Amdt. Dated December 14, 2005

Reply to Office action of September 15, 2005

01000 xx ... xx,

in which the bits designated xx may have arbitrary values.

Claim 8 (currently amended). [[A]] The method according to claim 7, wherein the higher-order section of the transformed modulus reads as follows:

01100 ... 00.

Claim 9 (currently amended). [[A]] The method according to claim 1, wherein ~~said~~ the step of transforming the modulus comprises randomization of the modulus so that the transformed modulus is randomized.

Claim 10 (currently amended). A processor for modular multiplication of a multiplicand by a multiplier within a cryptographic algorithm, in which a modulus is employed, wherein the multiplicand, the multiplier, and the modulus are parameters in the cryptographic algorithm, making use of using a multiplication look-ahead process and a reduction look-ahead process, comprising:

Applic. No.: 10/662,627

Amdt. Dated December 14, 2005

Reply to Office action of September 15, 2005

a ~~means~~ transformer for transforming the modulus into a transformed modulus ~~that is being~~ greater than the modulus by multiplying the modulus by a transforming number, the transforming number being calculated using the modulus such that with a predetermined fraction of the transformed modulus ~~having~~ has a higher-order digit with a first predetermined value ~~that is~~ followed by at least one lower-order digit having a second predetermined value;

a ~~means~~ processor for ~~iterative~~ iteratively working off the modular multiplication ~~making use of~~ using the multiplication look-ahead process and the reduction look-ahead process and utilizing the transformed modulus so as to obtain at the end of the iteration a transformed result for the modular multiplication, the predetermined fraction of the transformed modulus being used in the reduction look-ahead process; and

a ~~means~~ re-transformer for re-transforming the transformed result by modular reduction of the transformed result utilizing the modulus.

Claim 11 (currently amended). ~~[[A]]~~ The processor according to claim 10, comprising a host CPU and a coprocessor, said ~~means for transforming the modulus~~ transformer being arranged in the host CPU and said ~~means~~ processor for ~~iterative~~

Applic. No.: 10/662,627

Amdt. Dated December 14, 2005

Reply to Office action of September 15, 2005

iteratively working off ~~of~~ the modular multiplication being arranged in the coprocessor.

Claim 12 (currently amended). [[A]] The processor according to claim 11, wherein the host CPU is a short-number arithmetic-logic unit having a number of digits smaller than or equal to 64, and ~~wherein~~ the coprocessor is a long-number arithmetic-logic unit having a number of digits greater than or equal to 512.

Claim 13 (currently amended). [[A]] The processor according to claim 10, wherein the ~~means~~ processor for ~~iterative~~ iteratively the modular multiplication ~~comprises~~ includes a register for the transformed modulus and a register for an intermediate result of the modular multiplication.